

## **REMARKS**

### **STATUS OF THE CLAIMS**

Claims 53-55, 58 and 60-61 have been pending in the application.

Claims 53-55 and 58-60 are rejected under the judicially created doctrine of obviousness-type double patenting over claim 1 or 2 of US Patent No. 6,557,165.

The Office Action newly rejects claims 53-55, 58 and 60-61 under 35 USC 102(a) as being anticipated by Krishnamurthy, "Practical Reusable Unix Software," John Wiley & Sons, Inc. Copyright 1995 AT&T (hereinafter referred to as AT&T).

The Office Action maintains from the previous Office Action rejection of claims 53-55 and 58, and 60-61 under 35 U.S.C. 102(b) as being anticipated by Maxfield, "Designing With Objects", MachTech, Vol. 7, Issue No. 1, 1991.

According to the foregoing the claims are amended, and, thus, pending claims remain for reconsideration which is respectfully requested.

No new matter has been added.

### **DOUBLE PATENTING REJECTION**

Upon allowance of the pending claims, applicants may file a terminal disclaimer overcoming the non-statutory double patenting rejection.

### **35 USC 102 REJECTIONSS**

The Examiner maintains from the previous Office Action rejection of claims 53-55, 58, and 60-61 under 35 USC 102(b) as being anticipated by Maxfield, "Designing With Objects," MachTech, Vol. 7, Issue No. 1, 1991.

The Office Action newly rejects claims 53-55, 58 and 60-61 under 35 USC 102(a) as being anticipated by Krishnamurthy, "Practical Reusable Unix Software," John Wiley & Sons, Inc. Copyright 1995 AT&T (hereinafter referred to as AT&T).

The Office Action Response to Arguments in page 3 alleges the language of the claims is generic and abstract by reciting known object oriented component elements, and also implies

the claims might not be supported by the specification. The last sentence of the Response to Arguments suggests any component builder is a GUI associated with existing software to build and to rebuild components. The Applicants respectfully disagree, since the claims provide a benefit of replacing an existing software with an object by providing "***an object oriented programming component accessible by objects in a computer system***" that comprises (1) "***data related to existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software*** with the GUI in the computer system," and (2) "***as a component method for the existing software, a program code*** to control the computer system according to a process of ***receiving a message issued in another object***, and driving the existing software by ***issuing a GUI event of the existing software to the existing software, based upon the GUI data of the existing software, in response to the received message***, wherein the data and the component method together in combination with the existing software serve as one object."

Nevertheless, the independent claims 53, 55, 58 and 60, using claim 53 as an example, are amended to provide "data related to a non-object-oriented based software as existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system."

Further, for example, the present Application page 5, lines 2-8, and pages 175 (last paragraph) to page 183, expressly support the claims. In particular, FIG. 101 and page 178, line 4 to page 179, line 14 (claims 53, 60) and FIG. 103, page 180, line 5 to page 183, line 9 (claims 55, 58) support the claim amendments. Thus, the specification fully supports the language of the claims.

A prima facie case of anticipation of the claims as amended cannot be established based upon Maxfield, based upon the same arguments in the previous Amendment of April 11, 2007, which are not repeated here, namely there is no evidence that Maxfield's conventional discussion of designing an object oriented program discloses, expressly or inherently, the claimed "***an object oriented programming component accessible by objects in a computer system***" that comprises (1) "***data related to a non-object-oriented software as existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software*** with the GUI in the computer system," and (2) "***as a component method for the existing software, a program code*** to

control the computer system according to a process of ***receiving a message issued in another object***, and driving the existing software by ***issuing a GUI event of the existing software to the existing software, based upon the GUI data of the existing software, in response to the received message***,” providing a benefit of “the data and the component method together in combination with the existing software serve as one object to the objects.”

Further, AT&T Fig. 1.3 and pages 12-14, which is relied upon by the Office Action, discuss an application program’s structural components aimed at increasing reuse. AT&T page 14 discusses a component builder, such as a GUI builder, and that a programmer must attach a variety of application components, such as a GUI, to the style tool of an application program. However, this discussion is silent expressly or inherently on how a particular object-oriented object is defined. Further, AT&T FIG. 1.2 is a flowchart of the reusable software distribution process called ADVSOFT (see page 3), and does not relate the claimed object definition. Further, AT&T section 9, page 247+ and page 279 discuss conventional event action handling and process-centered software development environment, and do not relate the claimed object definition.

A prima facie case of anticipation of the claims as amended cannot be established based upon AT&T, because there is no evidence that AT&T’s discussion (pages 3-5) of a reusable software distribution process called ADVSOFT as applied to a software tool called YEAST discloses, expressly or inherently, the claimed ***“an object oriented programming component accessible by objects in a computer system”*** that comprises (1) ***“data related to a non-object-oriented software as existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system,”*** and (2) ***“as a component method for the existing software, a program code to control the computer system according to a process of receiving a message issued in another object, and driving the existing software by issuing a GUI event of the existing software to the existing software, based upon the GUI data of the existing software, in response to the received message,”*** providing a benefit of “the data and the component method together in combination with the existing software serve as one object to the objects.”

Withdrawal of the rejection of pending claims and allowance of pending claims is respectfully requested.

**CONCLUSION**

There being no further outstanding objections or rejections, it is submitted that the application is in condition for allowance. An early action to that effect is courteously solicited.

Finally, if there are any formal matters remaining after this response, the Examiner is requested to telephone the undersigned to attend to these matters.

Respectfully submitted,  
STAAS & HALSEY LLP

Date: February 13, 2008

By: / Mehdi D. Sheikerz /  
Mehdi D. Sheikerz  
Registration No. 41,307

1201 New York Avenue, NW, 7th Floor  
Washington, D.C. 20005  
Telephone: (202) 434-1500  
Facsimile: (202) 434-1501